

基于分组遗传算法的虚拟机放置方法

李淑英¹ 潘 亚¹ 费 薇² 徐 建²

(1. 商丘工学院 信息与电子工程学院 河南 商丘 476000; 2. 南京理工大学 计算机科学与工程学院 江苏 南京 210094)

摘 要: 为解决现有的虚拟机(VM)初始放置目标较为片面,仅考虑 1 个或者 2 个方面的问题,该文提出了 1 种面向多目标优化的 VM 初始放置方法。综合考虑了资源利用率、功率以及温度 3 方面因素。基于改进的分组遗传算法生成候选的 VM 放置方案。采用多目标模糊评估方法筛选出最佳放置方案。仿真实验结果表明,该文方法可以减少约 44% 的资源浪费、降低 3 kW 服务器运行功率。

关键词: 云计算; 虚拟化; 虚拟机放置; 分组遗传算法; 多目标优化; 资源利用率; 功率; 温度

中图分类号: TP312 **文章编号:** 1005-9830(2016)03-0322-06

DOI: 10.14177/j.cnki.32-1397n.2016.40.03.012

Virtual machine placement method based on grouping genetic algorithm

Li Shuying¹ Pan Ya¹ Fei Wei² Xu Jian²

(1. College of Information and Electronic Engineering, Shangqiu Institute of Technology, Shangqiu 476000, China;
2. School of Computer Science and Engineering, Nanjing University of Science and Technology,
Nanjing 210094, China)

Abstract: To solve the problem of existing virtual machine placement methods that the initial placement target is one-sided and only focuses on one or two optimization objects, a virtual machine initial placement method for multi-objective optimization is proposed here. Resource usage rate, system power and temperature are considered synthetically. Candidates of virtual machine placement solution are got based on an improved group genetic algorithm. The best virtual machine placement solution is selected by a multi-object fuzzy assessment algorithm. The simulation experiment results show that the proposed method can reduce the wasting of resources by 44% and server operation power by 3 kW.

Key words: cloud computing; virtualization; virtual machine placement; grouping genetic algorithm; multi-objective optimization; resource usage rate; power; temperature

收稿日期: 2016-01-08 修回日期: 2016-03-10

基金项目: 国家自然科学基金(61300053)

作者简介: 李淑英(1980-),女,硕士,讲师,主要研究方向: 软件工程、图形图像处理, E-mail: 879930173@qq.com;
通讯作者: 徐建(1979-),男,博士,副教授,主要研究方向: 虚拟化技术、数据挖掘, E-mail: dolphin.xu@njust.edu.cn。

引文格式: 李淑英,潘亚,费薇,等. 基于分组遗传算法的虚拟机放置方法[J]. 南京理工大学学报, 2016, 40(3): 322-327.

投稿网址: <http://zrxuebao.njust.edu.cn>

随着云计算的发展,研究人员致力于云数据中心资源管理研究。虚拟机(Virtual machine, VM)放置对于系统性能、能耗和确保服务质量有重要影响,是云计算领域的研究热点。VM 初始放置研究在 1 个空载的数据中心采用何种 VM 放置策略,这些放置策略需要考虑 VM 的资源需求,如 CPU、内存、IO 带宽等,物理主机是否具备容纳 VM 的能力,以及平台在功率、温度等方面的需求,并将这些需求转化为针对特定优化目标的约束条件,完成优化求解,从而得到初始放置方案。

现有的大部分研究工作仅关注云数据中心资源管理的 1 个或 2 个方面,如最大化资源利用率^[1-4]、最小化功耗^[5-8]或者均衡温度分布^[9-11]。在提高资源利用率方面,文献[3]为资源负载问题建立数学模型,然后利用蚁群优化算法实现了虚拟资源的均衡。文献[4]针对 VM 之间的依赖关系利用资源时间的互补性进行峰值分配放置,在提高 CPU 利用率的同时也节约了资源。在降低功耗方面,主要利用虚拟化进行工作负载整合,然后关闭零负载的物理机(Physical machine, PM),以节约能源^[5,6]。文献[7]提出了 1 种双目标优化模型,在优化 PM 资源利用率的同时使得 PM 的运行数量最小化,定义 VM 请求序列的矩阵集群以及 VM 放置的初始矩阵,基于矩阵变换算法生成最佳放置方案。文献[8]认为资源泄露会导致资源浪费以及功耗增加,因此给出了 1 种平衡算法避免资源泄露并提高资源利用率,减少了 PM 运行数量,从而降低了功耗。在均衡温度方面,服务器若因使用率太高而形成热点,可能会导致停机,如果数据中心的温度不平衡会增加冷却成本^[9]。文献[10]提出并实现了 1 种可以对处理温度紧急情况的管理方法进行预评估的基础设施,通过准确预测温度并给出相应的热管理方案,可以更有效地减少热点。文献[11]综合考虑温度和能耗 2 方面因素,提出了 1 种跨层热管理方法以及基于能量感知的 VM 放置方法,可以减轻采用虚拟化技术的云基础设施中的热异常,合理地解决 VM 和 PM 之间的映射问题,起到节能作用。

资源利用率、功耗和温度这 3 个因素是相互制约的。从节约能源的角度出发,应当将 VM 放置在尽量少的服务器上运行,这样可以关闭其它没有任何 VM 运行的 PM,从而减少能源成本,但是这样会引起温度失衡并且产生热点,导致需要更多的冷却成本、服务器寿命减少以及性能降低。

因此,研究 VM 放置问题需要考虑如何在这些目标之间达到平衡。本文提出的 VM 的初始放置方法中综合考虑了上述 3 个因素,分别提出了 3 个优化目标,通过改进分组遗传算法(Grouping genetic algorithm, GGA)在优化目标之间权衡,并且利用模糊逻辑的特点选择最佳方案,使得算法具有提高资源利用率、降低服务器运行功率以及减少热点的综合性能。

1 问题描述和建模

本文的 VM 初始放置方法考虑以下几个因素:

(1) 资源浪费。为了平衡不同资源维度的使用情况,采用如下符号计算资源浪费的潜在成本,其中 R_i 表示第 i 维剩余资源量占该维资源总量的百分比。剩余资源最少的维度用下标 k 表示,则服务器的剩余资源浪费总量 W 可表示为每维剩余资源百分比与 R_k 之差的总和 $W = \sum_{i \neq k} [R_i - R_k]$ 。由此可见, W 越大,资源浪费越多。

(2) 功耗。使用 1 种常用的线性功率模型^[12]估计功耗的大小。为了节约能源,当服务器上没有任何 VM 运行即空载时,就将它关闭或者调至低功率状态。服务器的总运行功率 P 如下

$$P = \sum_j P_j \quad P_j = \begin{cases} p_1 + p_2 U_j^{\text{CPU}} & U_j^{\text{CPU}} > 0 \\ 0 & \end{cases} \quad (1)$$

式中: P_j 表示第 j 台 PM 的功耗, U_j^{CPU} 表示第 j 台 PM 的 CPU 利用率, p_1 、 p_2 为参考系数。

(3) 温度。服务器若因使用率太高而形成热点,可能会产生破坏性的停机时间。根据热传递和相移电路现象之间的对偶性,热传递和相移电路可以用来模拟处理器的稳态温度。服务器温度计算如下

$$t = PR + t_{\text{amb}} \quad (2)$$

式中: P 表示功耗, R 表示热阻, t_{amb} 表示环境温度。式(1)和式(2)温度与主机的 CPU 负载有关,利用功耗和 CPU 活动之间的线性关系可以估计出温度的大小。数据中心热管理的目的在于减少热点,将温度保持在安全运行范围内,保持数据中心服务器的温度平衡。

VM 初始放置问题可以描述如下:数据中心有 M 台 PM,现有 N 台 VM 发起请求,第 i 台 VM 发起资源请求,本文只考虑 CPU 和内存。具体描

述符号如表 1 所示。

表 1 多目标 VM 放置问题形式化符号描述表

符号	描述
M	物理服务器的数量
N	发出请求的 VM 数量
$[c_j^{\text{cpu}} \ c_j^{\text{mem}}]$ $j=1, \dots, M$	第 j 台服务器的资源容量向量
$[r_i^{\text{cpu}} \ r_i^{\text{mem}}]$ $i=1, \dots, N$	第 i 台 VM 的资源请求向量
$a_{ij} \in [0, 1]$	如果 $a_{ij}=1$, 表示第 i 台 VM 分派在第 j 台服务器上
W_j	第 j 台服务器的资源浪费量
P_j	第 j 台服务器消耗的功率
t_j	第 j 台服务器的温度
U_{cpu}	CPU 利用率
U_{mem}	内存利用率

3 个目标函数可表示为 $\min \sum_{j=1}^M W_j$ 、 $\min \sum_{j=1}^M P_j$ 、 $\min \max(t_j)$, 依次表示最小化资源浪费量、最小化运行功率、最小化温度峰值。约束条件可表示为: $\sum_{i=1}^N r_i^{\text{cpu}} a_{ij} < c_j^{\text{cpu}}$ 、 $\sum_{i=1}^N r_i^{\text{mem}} a_{ij} < c_j^{\text{mem}}$ 、 $\sum_{j=1}^M a_{ij} = 1$ 。其中 $i \in \{1, 2, \dots, N\}$ 、 $j \in \{1, 2, \dots, M\}$ 。前 2 个约束条件表示每个 PM 被分配的资源大小不能超过其本身具有的资源总量, 第 3 个约束条件保证每台 VM 只会分配给 1 台 PM。

2 基于模糊评估的 GGA

2.1 编码

遗传算法 (Genetic algorithm, GA) 是 1 种随机的基于种群搜索的技术, 每组经过编码的字符串称为染色体, 每个染色体代表解空间的 1 个解, 每个解都有 1 个适应度, 代表了这个解与最优化目标相比的优劣程度。种群候选方案会逐渐向最优化方案变化。传统的 GA 在分组问题如装箱问题等上执行不佳, 因为传统的染色体编码并不能体现箱子的结构。GGA 改进了 GA 的内容以适应分组问题的结构。传统 GA 的染色体编码主要针对对象而非分组, GGA 使用了 1 种特殊编码方案将基因分组后的染色体作为编码对象。在分组问题中, 目标函数基于组而不是基于独立的个体, 因此, GGA 中采用组作为编码对象。图 1 为基于分

组编码的 1 个具体例子。9 台 VM 被划分成 3 组, 分别放置在 3 台 PM 上, 这种编码方式将算法从面向 VM 转变为面向 PM, 可以很好地体现将 VM 放置在 PM 上运行的结构。

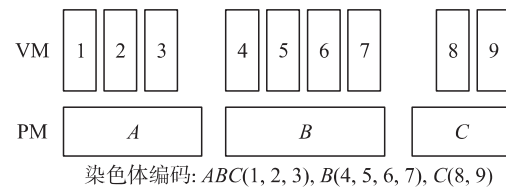


图 1 基于分组的染色体编码图

2.2 交叉和变异

交叉产生子代的具体过程如图 2 所示。由于采用分组编码并且是随机选择插入, 因此在形成的子代中有些 VM 可能会多次出现在不同的 PM 中, 如果出现这种情况则需要将包含这些相同 VM 的组 (PM) 从子代中删除, 这个删除步骤带来的问题是可能会出现没有被分配的 VM。针对这个问题, GGA 采用启发式算法如最先匹配算法, 重新插入未分配的 VM。但是, 这种交叉操作在 VM 初始化放置问题中效果较差, 因为子代的生成完全依靠随机选择、插入, 较少次数的尝试不可能获得较好的结果。因此, 本文提出 1 种基于排名的交叉操作, 能够更高效地继承父代的优秀基因。采用资源使用效率、能耗效率、热效率等 3 种效率对每个独立的组 (装有 VM 的 PM) 进行评估, 这 3 种效率分别对应前文设计的 3 个优化目标。

资源使用效率 (E_{resource}) 反映不同类型的资源的利用程度, 本文的目标是充分利用所有维度上的资源, 若只考虑 CPU 和内存, 则资源使用效率如下

$$E_{\text{resource}} = U_{\text{cpu}} U_{\text{mem}} \quad (3)$$

功耗效率 (E_{power}) 反应单位功耗下可以产生的有用工作量, 会随着 CPU 利用率的增加而单调递增, 当 CPU 利用率为 100% 时功耗效率达到峰值, 功耗效率如下

$$E_{\text{power}} = \frac{w_{\text{orkload}}}{p_{\text{ower}}} = \frac{U_{\text{cpu}}}{p_1 + p_2 U_{\text{cpu}}} (p_1 + p_2) \quad (4)$$

式中: 参数 $p_1 + p_2$ 将 E_{power} 控制在 $[0, 1]$ 内。

热效率 (E_{thermal}) 在安全范围内随着 CPU 温度的升高而缓慢减小, 当 CPU 温度超出安全范围时则热效率急剧下降, 热效率如下

$$E_{\text{thermal}} = \frac{1}{1 + e^{T - T_s}} \quad (5)$$

式中: T_s 为参数。这 3 个效率的值都在 0 到 1 之间, 染色体使用这 3 个效率的平均值作为评价函

数, 如下

$$F_e = \text{avg}(E_{\text{resource}} E_{\text{power}} E_{\text{thermal}}) \quad (6)$$

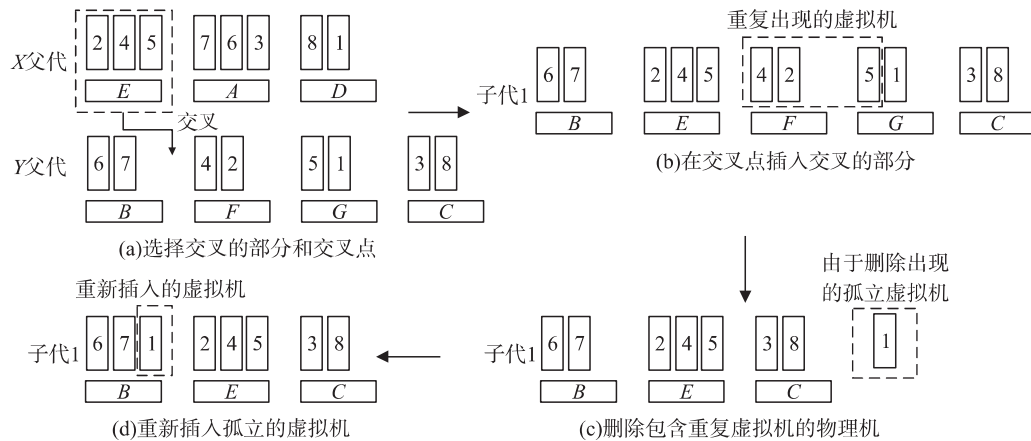


图 2 交叉操作产生子代过程图

若 1 台 VM 同时出现在多个 PM 中, 则将涉及的所有 PM 按照评价函数从大到小排序, 选择评价函数最大的 PM, 并将这个 VM 从其他 PM 中删除。因删除操作而剩余的 VM 按照最先匹配原则插入 PM。通过这种方式, 子代可以很好地继承父代的优秀部分, 从而更快地生成更好的方案。

GGA 变异依旧是以前 (PM) 为对象的。一般通过随机删除 PM 的方式实现变异, 但是盲目的删除和插入会使群体更加分散。本文采用变异操作删除评价函数最小的 PM, 能够保证变异带来优化的效果, 这些被删除的 PM 上的 VM 依旧按照最先匹配原则插入其他 VM。

2.3 模糊评估

模糊逻辑是 1 种用来处理不确定、不精确或者定性决策问题的工具。在布尔逻辑中, 元素 x 属于集合 A 或者不属于集合 A ; 在模糊逻辑中, 元素 x 部分地属于集合 A 或者不属于集合 A , 属于集合 A 的程度称为隶属度。VM 放置尝试将几个冲突的目标同时最小化。为了使用 GGA 解决多目标问题, 用来选择新一代候选方案的适应度函数必须能够反映所有的目标, 并且基于模糊逻辑系统评估 GGA 产生的候选解决方案。

本文提出的 VM 放置策略的目标是最小化资源浪费、最小化功耗以及最小化最高温度。采用 3 种语言变量: 资源浪费 w , 功耗 P , 温度 t 。每个语言变量的取值对应 1 个模糊集合, 即 s_w 、 l_p 以及 l_t 。这些模糊集合的隶属度函数是关于变量值的递减函数, 例如变量的温度越高, 则该变量对于低温模糊集合的隶属度就越低。搜索算法试图找到与每个单独的目标都最接近的解

决方案, 因此, 每个解决方案的评估都可以表示成这样的模糊规则: 如果解决方案 x 拥有少量资源浪费 s_w 、低功耗 l_p 以及低温 l_t , 则 x 是 1 个好的解决方案。

在模糊集合 s_w 、 l_p 、 l_t 中, 隶属度最高的方案即为最理想的解决方案, 这里使用有序加权平均模糊算子, 则上文的模糊规则如式 (7) 所示

$$\mu(x) = \beta \min(\mu_w(x), \mu_p(x), \mu_t(x)) + (1-\beta) \text{avg}(\mu_w(x), \mu_p(x), \mu_t(x)) \quad (7)$$

式中: β 为参数, $\mu_w(x)$ 、 $\mu_p(x)$ 、 $\mu_t(x)$ 分别代表方案 x 在 s_w 、 l_p 、 l_t 定义的模糊集合中的隶属度, $\mu(x)$ 是方案 x 在最理想方案模糊集合中的隶属度。

模糊集合 s_w 、 l_p 、 l_t 的隶属度函数都是线性递减函数, 下文计算隶属度函数的上界和下界。 R_{cpu} 和 R_{mem} 分别表示所有 VM 请求的 CPU 总量和内存总量, $C_{\text{cpu}}^{\text{max}}$ 和 $C_{\text{mem}}^{\text{max}}$ 分别表示资源量最大的 PM 的 CPU 容量和内存容量, $C_{\text{cpu}}^{\text{min}}$ 和 $C_{\text{mem}}^{\text{min}}$ 分别表示资源量最小的 PM 的 CPU 容量和内存容量。在理想状况下, 希望能够用最少的服务器来运行所有的 VM, 因此假设 VM 可以被分割且每台服务器都充分被利用, 则需要使用的服务器最小数量 $m_{\text{min}} = \max(R_{\text{cpu}}/C_{\text{cpu}}^{\text{max}}, R_{\text{mem}}/C_{\text{mem}}^{\text{max}})$ 和最大数量 $m_{\text{max}} = \min(M, N)$, 功耗的下界为 $P_{\text{lower}} = m_{\text{min}}(p_1 + p_2)$ 和上界 $P_{\text{upper}} = m_{\text{max}}p_1 + (R_{\text{cpu}}/C_{\text{cpu}}^{\text{min}})p_2$ 。若在能容纳 VM i 的 PM 之中资源量最小的是 PM j , 则总资源浪费的下界 $W_{\text{lower}} = |\sum_i (r_i^{\text{cpu}}/C_{\text{cpu}}^{\text{max}}) - \sum_i (r_i^{\text{mem}}/C_{\text{mem}}^{\text{max}})|$ 和上界 $W_{\text{upper}} = \sum_i |r_i^{\text{cpu}}/c_j^{\text{cpu}} - r_i^{\text{mem}}/c_j^{\text{mem}}|$ 。

本文的温度模型采用式 (2) 所示的线性模型,

则 CPU 温度的下界 $t_{\text{lower}} = t_{\text{amb}}$ 和上界 $t_{\text{upper}} = (p_1 + p_2) R + t_{\text{amb}}$ 。所以,假设 x_w, x_p, x_t 分别表示方案 x 中的资源浪费、功耗、CPU 温度,则 s_w, l_p, l_t 对于方案 x 的隶属度函数分别为 $\mu_w(x) = \frac{W_{\text{upper}} - x_w}{W_{\text{upper}} - W_{\text{lower}}}$, $\mu_p(x) = \frac{P_{\text{upper}} - x_p}{P_{\text{upper}} - P_{\text{lower}}}$ 和 $\mu_t(x) = \frac{t_{\text{upper}} - x_t}{t_{\text{upper}} - t_{\text{lower}}}$ 。最后根据式(7)可以对方案 x 给出评估,拥有最高隶属度 $\mu(x)$ 的方案即为最佳方案,满足所有的目标。

3 仿真实验

实验采用 CloudSim 进行仿真,具体实验参数如表 2 所示。实验中运行功率和温度模型均采用线性模型进行模拟仿真,运行功率随 CPU 利用率变化的情况会因为服务器的类型而不同,因此设置了 3 种不同的运行功率: $P = 110 + 60U_{\text{cpu}}$, $P = 110 + 65U_{\text{cpu}}$ 和 $P = 110 + 70U_{\text{cpu}}$,并根据服务器类型分配。温度设置同样有 3 种情况: $t = 15 + 30U_{\text{cpu}}$, $t = 15 + 35U_{\text{cpu}}$ 和 $t = 15 + 40U_{\text{cpu}}$,随机分配。

将本文提出的基于模糊评估的多目标分组遗传算法与其他 3 种算法相比较。GGA 交叉过程中子代的生成完全依靠随机选择、插入,通过随机

删除 PM 的方式来实现变异。仅考虑功耗的分组遗传算法(Single objective grouping genetic algorithm based on power, SGGA-P)使用单目标的 GGA。仅考虑温度的分组遗传算法(Single objective grouping genetic algorithm based on temperature, SGGA-T)使用单目标的 GGA。多目标分组遗传算法(Multi objectives grouping genetic algorithm, MGGA)使用 GGA,从资源浪费、功耗、温度这 3 个方面计算适应度。

从图 3 可以看出,SGGA-P 的温度最高,因为这种算法倾向于将 VM 合并到较少数量的服务器,资源利用率越高,服务器温度越高,但是带来的好处是 SGGA-P 的功耗最小。SGGA-P 算法的资源浪费也比较少,因为这种算法从功耗的角度考虑,会尽量充分使用资源。SGGA-T 算法的温度最低,因为这种算法趋向于将 VM 均匀分配给服务器,资源利用率较低,因此 CPU 温度也较低。MGGA 算法的功耗、温度以及资源浪费都相对较低,因为它综合考虑了这 3 种因素,拥有很好的平衡能力。MGGA 算法的适应度值最高,这表明其综合能力最佳。

表 2 实验环境参数表

VM 请求数量	VM 对 CPU 请求的大小集合	VM 对内存请求的大小集合	服务器数量	服务器的 CPU 大小集合	服务器的内存大小集合	$t_s / ^\circ\text{C}$	β
250	{0.25 0.5 1 1.5 2 2.5 3 4}	{0.25 0.5 1 1.5 2 2.5 3 4}	100	{4 8 16}	{4 8 16}	60	0.5

4 结束语

本文提出了 1 种综合考虑资源浪费、运行功率和温度等因素的 VM 初始放置方案。设计了减小化资源浪费、降低运行功率以及降低温度峰值 3 个优化目标,采用了基于模糊评估的 GGA 进行优化求解。首先,根据 3 个目标改进了 GGA 的交叉、变异过程;然后,运用模糊逻辑评估遗传算法生成候选方案;最后,利用隶属度作为评价函数从而选择出最佳方案。仿真结果表明,与考虑单目标的模糊评估 GGA 以及传统的 GGA 相比,该算法具有良好的综合性能,可以在资源浪费、运行功率以及温度这 3 方面大大改善 VM 初始放置效果。在今后的工作中,需要进一步提高温度和功

率模型的精确度,另外,还可以考虑影响 VM 初始放置的其它因素,比如服务层协议(Service-level agreement, SLA)、网络带宽等。

参考文献:

- [1] Dalvandi A, Gurusamy M, Chua K C. Power-efficient resource-guaranteed VM placement and routing for time-aware data center applications [J]. Computer Networks, 2015, 88(C): 249–268.
- [2] Chen M T, Hsu C C, Kuo M S, et al. GreenGlue: Power optimization for data centers through resource-guaranteed VM placement [C]//IEEE International Conference on Internet of Things(iThings) and IEEE Green Computing and Communications(GreenCom) and IEEE Cyber, Physical and Social Computing(CPSCom). Tai-

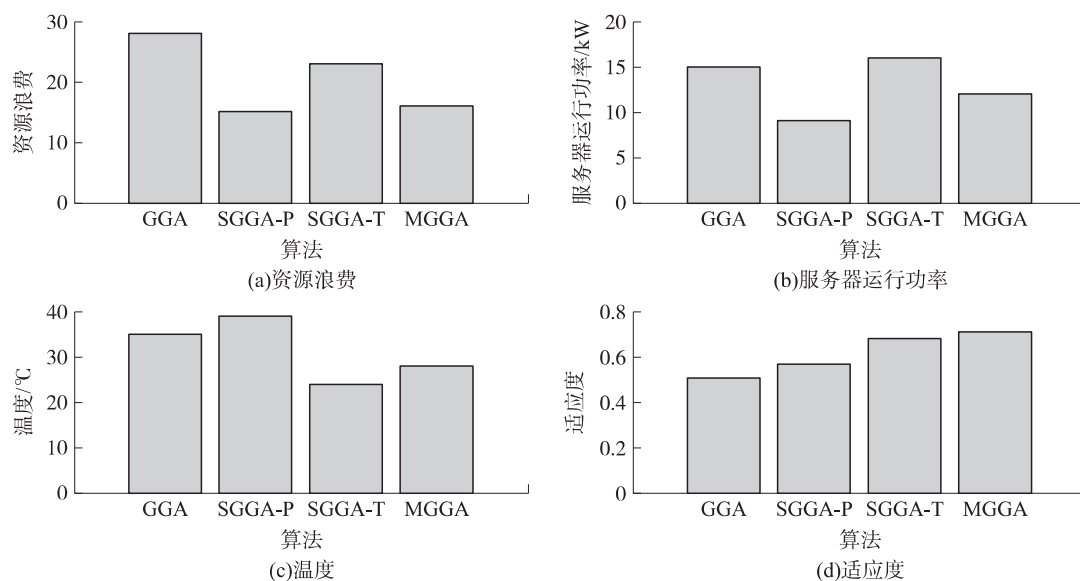


图 3 4 种 GGA 性能对比图

pei ,China: IEEE 2014: 510–517.

- [3] 张牧. 云计算和多维 QoS 环境中基于蚁群优化算法在虚拟机资源负载均衡问题中的研究 [J]. 计算机科学 2013 40(11A): 60–62.

Zhang Mu. Research of virtual machine load balancing based on ant colony optimization in cloud computing and multi-dimensional QoS [J]. Computer Science , 2013 40(11A): 60–62.

- [4] 潘飞, 蒋从锋, 徐向华, 等. 负载相关的虚拟机放置策略 [J]. 小型微型计算机系统 2013 34(3): 520–524.

Pan Fei ,Jiang Congfeng ,Xu Xianghua ,et al. Placement strategy of virtual machines based on workload characteristics [J]. Journal of Chinese Computer Systems , 2013 34(3): 520–524.

- [5] 秦启飞, 王世振, 袁翔, 等. 云环境下基于 CROTS 算法的虚拟机放置策略 [J]. 计算技术与自动化, 2015 34(1): 105–110.

Qin Qifei ,Wang Shizhen ,Yuan Xiang ,et al. Chemical reactive optimization for VM consolidation in cloud computing environment [J]. Computing Technology and Automation 2015 34(1): 105–110.

- [6] 吴毅华, 曹健, 李明禄. 云计算环境下基于需求预测的虚拟机节能分配方法研究 [J]. 小型微型计算机系统 2013 34(4): 778–782.

Wu Yihua ,Cao Jian ,Li Minglu. Energy efficient allocation of virtual machines in cloud computing environments based on demand forecast [J]. Journal of Chinese Computer Systems 2013 34(4): 778–782.

- [7] Sun Meng ,Gu Weidong ,Zhang Xinchang ,et al. A matrix transformation algorithm for virtual machine placement in cloud [C] //2013 12th IEEE International

Conference on Trust ,Security and Privacy in Computing and Communications (TRUSTCOM 2013) . Melbourne ,VIC ,Australia: 2013: 1778–1783.

- [8] Li Xin ,Qian Zhuzhong ,Chi Ruiqing ,et al. Balancing resource utilization for continuous virtual machine requests in clouds [C] //MIS'12 Proceedings of the 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. Washington ,DC ,USA: IEEE Computer Society ,2012: 266–273.

- [9] Wang Lizhe ,Khan S U ,Dayal J. Thermal aware workload placement with task-temperature profiles in a data center [J]. The Journal of Supercomputing ,2012 ,61(3): 780–803.

- [10] Ramos L ,Bianchini R. C-Oracle: Predictive thermal management for data centers [C] //IEEE 14th International Symposium on High Performance Computer Architecture. Salt Lake City ,UT ,USA: IEEE ,2008: 111–122.

- [11] Rodero I ,Viswanathan H ,Lee E K ,et al. Energy-efficient thermal-aware autonomic management of virtualized HPC cloud infrastructure [J]. Journal of Grid Computing 2012 ,10(3): 447–473.

- [12] Elnozahy E N ,Kistler M ,Rajamony R. Energy-efficient server clusters [J]. Lecture Notes in Computer Science 2003 2325: 179–197.